CLOUD COVER/LAYERS

VISIBLE/INFRARED IMAGER/RADIOMETER SUITE
ALGORITHM THEORETICAL BASIS DOCUMENT

Version 3: May 2000

Duane Apling
Maureen Cianciolo

RAYTHEON SYSTEMS COMPANY
Information Technology and Scientific Services
4400 Forbes Boulevard
Lanham, MD 20706

SRBS Document #: Y2392

EDR: Cloud Cover and Layers (40.4.2)

Doc No: Y2392

Version: 3

Revision: 0

	FUNCTION	NAME	SIGNATURE	DATE
Prepared by	EDR Developer	D. APLING M. CIANCIOLO		4/17/00
Approved by	Relevant IPT Lead	G. HIGGINS		4/18/00
Approved by	Chief Scientist	P. ARDANUY		
Released by	Program Manager	H. BLOOM		

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

GLOSSARY OF ACRONYMS

ATBD	Algorithm Theoretical Basis Document
AVHRR	Advanced Very-High-Resolution Radiometer
CSSM	Cloud Scene Simulation Model
DISORT	Discrete Ordinate Radiative Transfer Model
DMSP	Defense Meteorological Satellite Program
DOC	U.S. Department of Commerce
DOD	U.S. Department of Defense
EDR	Environmental Data Record
FDTD	Finite-Difference Time Domain Method
FIRE	First ISCCP Regional Experiment
GTC	Graph Theoretic Clustering
HCS	Horizontal Cell Size
HITRAN	High-Resolution Transmission Molecular Absorption Database
IPT	Integrated Project Team
IR	Infrared
ISCCP	International Satellite Cloud Climatology Project
IWC	Ice Water Content
LMT	Layer Merge Test
LOWTRAN	Low-Resolution Transmission Model
MODIS	Moderate Resolution Imaging Spectroradiometer
MODTRAN	Moderate-Resolution Atmospheric Transmittance and Radiance Model
NOAA	National Oceanic and Atmospheric Administration
NPOESS	National Polar-Orbiting Operational Environmental Satellite System
OLE	Obscured Layer Extension
POES	Polar-Orbiting Operational Environmental Satellite
SNR	Signal-Noise Ratio
SPR	Statistical Pattern Recognition
SRD	Sensor Requirement Document
TBD	To be determined by contractor
TBR	To be resolved
VIIRS	Visible/Infrared Imager/Radiometer Suite

ABSTRACT

Total cloud cover and layered cloud structure are important components of a nephanalysis. Many operational users of cloud data require analysis of the extent, type, and physical characteristics of vertically distributed cloud layers. For example, in-flight aircraft refueling has stringent requirements for cloud-free visibility between aircraft at flight altitude. Icing specification and forecasts depend on accurate initial depiction of the constituent particle sizes and state (liquid or frozen) of clouds at specific altitudes. Many other uses, such as accurate prediction of lines-of-sight for aerial reconnaissance, depend on accurate classification of clouds into the classic cloud families. These diverse needs are best met by a general algorithm called Graph Theoretic Clustering (GTC), which uses local relationships among pixels to connect them in clusters.

This algorithm is capable of working well with many different types of data and identifies layers that are locally accurate and consistent across longer distances. Input parameters required include pixel-level information such as the Cloud Top Height and/or Temperature/Pressure, the Effective Particle Size, and the Cloud Optical Depth Environmental Data Records (EDRs). The pixel-level cloud mask and confidence flags are used as well.

1.0 INTRODUCTION

The Cloud Cover/Layers Algorithm ingests pixel-level Environmental Data Records (EDRs) from other supporting algorithms (e.g., cloud top heights, optical depths, effective particle sizes, and the cloud mask of the Visible/Infrared Imager/Radiometer Suite [VIIRS]) and performs a statistical analysis to identify distinct sets of cloud layers.

The Cloud Layer algorithm uses a number of Statistical Pattern Recognition (SPR) techniques, and requires some knowledge of jargon common to that discipline. A list of definitions and simple examples of the SPR terminology used in this document follows.

- *Density*. Density is the probability mass per unit parameter space volume. The parameter space may include dimensions which are spatial (e.g., space and/or time) and parametric (e.g., temperature, optical depth, or particle size). Density is commonly estimated from sample data using *Density Estimation* techniques. These techniques commonly use weighted sums of counts of data points falling into a sub-volume of the parameter space.
- *Graph Tree.* A graph tree is a biological analogy for the structure of a directed graph. The analogy surfaces from the appearance of a graph. Links start from outer nodes (*leaf* nodes) and converge to inner nodes (*stem* nodes). Inner nodes link to *branch* nodes which in turn, link to the *root* node.
- *Link.* A link is a relationship between two data points (also called *nodes*) establishing connectivity from one node to the other. The direction is one-way; one node connecting to another does not imply the other connects to the first. If two-way connectivity is required, two links are drawn, one in each direction.
- *Root.* The central node to which all links converge.
- *Branch.* The next set of nodes out from the root. These connect directly to the root node.
- *Stem.* Intermediary nodes out from the branch nodes.
- *Leaf.* The outermost nodes.
- *Predecessor.* A node which connects to another node is called the *predecessor* of the other node.
- *Prior Distribution.* The population distribution from which a sample is known to have been drawn.
- *Significance.* The probability of a sample having a given statistical value. Extreme probability levels indicate *incompatibility* with the null hypothesis.
- *Compatibility.* Statistical evidence is compatible with a hypothesis when the improbability of the statistics given the hypothesis is accepted is not subjectively high.
- *Bhattacharyya Distance.* The *Bhattacharyya Distance* is a statistical measure of separation between two multivariate Gaussian distributions. For example, two standard normal

distributions with unit variance, but separated in the mean by 1 unit have a Bhattacharyya Distance of D=1/8. Separating them further, out to two units, multiplies the distance by four to D=1/2. If the two distributions are equally likely prior distributions, then the approximate error rate from assigning samples to the closest distribution is 0.5*exp(-D).

1.1 PURPOSE

The Cloud Layer Algorithm is intended to meet the requirements for cloud layer specification defined in the Visible/Infrared Imager/Radiometer Suite (VIIRS) Sensor Requirements Document (SRD). This document describes algorithms that will be used to determine the layered structure of cloud cover using a variety of VIIRS EDRs. These EDRs include cloud top height and optionally optical depth and effective particle size. The algorithm also requires the results of the VIIRS cloud mask, which is not a formal EDR as specified by the SRD. This document contains a description of data flow, the layering algorithms and their statistical basis, sensitivity studies, and implementation considerations.

1.1.1 Definition of a Distinct Cloud Layer

Clouds are not distinct physical objects with inherent structure and form. They are best described as perceptual entities, that is, entities whose distinctiveness is based on the judgments of observers. For a collection of perceived clouds to form a layer, those clouds must share characteristics perceived by observers to be substantially different from those of other distinct layers. The obvious models of cloud layers (the slab and grid models, described below) both fall short in matching the breadth of perceptual characteristics of cloud layers. For example, a single cirrus layer may change dramatically in height, optical depth, and particle size, and yet be “obviously” a single entity to a ground observer. The observer may note physical connectivity and structured texture as the defining elements of a single cirrus layer, neither of which is easily captured by an objective algorithm and layer model. The vertical grid may accurately model the spatial distribution of cloud particles and their physical characteristics, but not describe them adequately as fully connected. The slab model may be able to model the cloud as a single connected layer, but be a rather poor model of the variation of the physical properties within the layer. Clearly, any objective model of cloud layers following these two patterns will not fully capture the cloud layer structure.

1.1.1.1 The Grid Model

The Grid Model directly describes the vertical distribution of cloud parameters for each horizontal cell at a series of vertically spaced intervals. For example, within a horizontal cell, vertical grid points may be established in 100 meter increments up to 30km, for a total of 300 intervals. At each grid point, the cloud parameters valid at the associated altitude are listed.

1.1.1.2 The Slab Model

The Slab Model describes the vertical extent of clouds at each horizontal cell. For a given number of existing layers, the altitude of the bottom and top of each layer is given. For each layer, a set of descriptive parameters (e.g., optical depth, particle size, cloud top temperature, etc.) are given as well.

1.1.2 Statement of the Cloud Layer Discernment Problem

Given a collection of pixels and corresponding recovered physical parameters, identify and describe a vertical distribution of cloud cover that is compatible with the measured scene. The ideal distribution is the only compatible description, or at least the most compatible description, that can be found.

1.1.3 Overview of the Cloud Cover/Layers Algorithm

The Cloud Cover/Layers Algorithm operates on the VIIRS cloud mask and assigns cloudy pixels to various layers by selecting small groups of pixels that share common physical parameters. These include cloud top height and optically cloud optical depth and effective particle size. These three parameters allow distinction of clouds at different altitudes and of different types of clouds. The resulting small layers are statistically merged as needed into a layered structure (sorted by cloud top height). A probabilistic model of the co-occurrence of clouds in different layers is then applied to estimate the amounts of coverage of obscured layers.

1.2 SCOPE

This Algorithm Theoretical Basis Document (ATBD) details an algorithm to reconstruct, on a pixel basis, the vertical distribution of cloud cover within small geographic cells from recovered physical parameters. The selected algorithm fits three dimensional cloud cover to a vertically ordered slab model of cloud structure within each aggregated horizontal cell to meet and exceed the threshold requirements for this EDR. Section 1 provides an overall introduction to this document, covering its purpose and scope and identifying relevant VIIRS documents. Section 2 describes the objectives of cloud layer retrievals, identifies the relevant characteristics of the VIIRS instrument, and summarizes retrieval strategy. Section 3 describes the algorithms itself in detail and Section 4 gives the assumptions and limitations underlying its use. Section 5 lists relevant textual references.

1.3 VIIRS DOCUMENTS

None.

1.4 REVISIONS

PR-08923-04-02, Version 1 Revision 0, Annotated Abstract, June 10, 1998.

PR-08923-04-02, Version 1, Revision 0.1, Annotated Outline, August 15, 1998.

Y2392, Version 1, Revision 2, Cloud Cover/Layers ATBD, October 1998.

Y2392, Version 1, Revision 3, Cloud Cover/Layers ATBD, March 1999.

Y2392, Version 2, Revision 0, Cloud Cover/Layers ATBD, June 1999.

Y2392, Version 3, Revision 0, Cloud Cover/Layers ATBD, May 2000.

2.0 EXPERIMENT OVERVIEW

2.1 Objectives of Cloud Layer Retrievals

The Cloud Cover/Layers Algorithm will be developed to meet SRD requirements. This algorithm will use EDRs that have been retrieved for each image pixel rather than computed from horizontally aggregated EDRs (i.e., EDRs for horizontal cells). The objective is to identify the vertical structure of clouds consistent with the pixel-level EDRs within each horizontal (aggregation) cell over a VIIRS image.

The SRD provides the following definition for cloud cover/layers:

> *"Cloud cover/layers consists of two data products:*
>
> *(a) fractional cloud cover, defined (TBR) as the fraction of a given area on the Earth's surface for which a locally normal line segment extending between two given altitudes intersects a cloud, and*
>
> *(b) a binary (cloudy/not cloudy) map at the pixel level indicating the pixels which are deemed to contain clouds. The detection criterion for cloudiness at the pixel level is TBD.*
>
> *As a threshold, fractional cloud cover is required for up to four layers of the atmosphere between the surface and an altitude of 20 km. As an objective, cloud cover is required for contiguous, 0.1 km thick layers at 0.1 km increments in altitude, from the surface of the Earth to an altitude of 30 km.."*

The SRD requirements for cloud cover/layers are set out in Table 1. *This ATBD addresses only the fractional cloud cover product.*

2.2 INSTRUMENT CHARACTERISTICS

The characteristic of the VIIRS instrument that most affects the cloud cover/layers product is the oblique viewing geometry resulting from the orbital plane of the NPOESS platform and the required width of swath (3000 km [TBR]).

The VIIRS instrument will have pixel footprints that are ellipsoidal and off-nadir, and that show degraded spatial resolution relative to the nadir case. The lower pixel resolution off-nadir may impair the cloud cover/layers product by offering fewer samples per horizontal cell.

Additionally, the off-nadir pixels will have been sensed at oblique look angles. In this case, adjacent cloud elements will tend to obscure cloud-free regions between them, increasing the measured amount of cloud in each horizontal cell. Finally, the edges and sides of larger clouds will be sensed more preferentially, and these may show as spurious small cloud layers in the analysis.

Table 1. SRD requirements for cloud cover/layers.

SRD Para. No.		Thresholds	Objectives
	a. Horizontal Cell Size		
V40.4.2-11	1. Fractional cloud cover	25 km	2 km
V40.4.2-12	2. Binary map	pixel size	(TBD)
V40.4.2-2	b. Horizontal Reporting Interval	(TBD)	(TBD)
	c. Vertical Cell Size	N/A	N/A
V40.4.2-3	d. Vertical Reporting Interval (fractional cloud cover)	Up to 4 layers	0.1 km
V40.4.2-4	e. Horizontal Coverage	Global	Global
V40.4.2-5	f. Vertical Coverage	0 - 20 km	0 - 30 km
V40.4.2-6	g. Measurement Range	0 - 1.0	0 - 1.0
V40.4.2-14	1. Fractional cloud cover	0 – 1.0	0 – 1.0
V40.4.2-15	2. Binary map	Cloudy/not cloudy	Cloudy/not cloudy
V40.4.2-7	h. Measurement Accuracy (fractional cloud cover)	0.1	0.05
V40.4.2-8	i. Measurement Precision (fractional cloud cover)	0.15	0.025
V40.4.2-13	n. Probability of Correct Typing (binary map)	> (TBD) at (TBS) % confidence level	> (TBD) at (TBS) % confidence level
V40.4.2-9	j. Mapping Uncertainty	4 km	1 km
	k. Maximum Local Average Revisit Time	6 hrs	4 hrs
	l. Maximum Local Refresh	(TBD)	(TBD)
V40-4.2-10	m. Minimum Swath Width (All other EDR thresholds met)	3000 km (TBR)	(TBD)

2.3 RETRIEVAL STRATEGY

The strategy is to use unsupervised clustering to group pixels into distinct layers. Each distinct layer represents a slab entity in the vertically stacked slab model of cloud structure. Pixels are grouped into layers on the basis of statistical similarity and geographic proximity. Populations of member pixels are used to infer unobscured cloud amount for each layer, and a statistical inference to the total amount of cloud (both obscured and unobscured) is made.

This strategy relies on:

- Similar physical properties of pixels that are members of the same layer.
- Statistical and probabilistic modeling to combine similar small layers into single consistent, substantial layers.
- A vertical slab model of cloud layers.
- A within-horizontal-cell correlation model to extend obscured layers beneath higher obscuring layers.

3.0 ALGORITHM DESCRIPTION

The Cloud Cover/Layers Algorithm operates on the VIIRS cloud mask and assigns cloudy pixels to various layers by selecting small groups of pixels that share common physical parameters. These parameters include cloud top height and optionally cloud optical depth and effective particle size. These three parameters allow distinction of clouds at different altitudes and of different types of clouds. The resulting small layers are statistically merged as needed into a layered structure (sorted by cloud top height). A probabilistic model of the co-occurrence of clouds in different layers is then applied to estimate the amounts of coverage of obscured layers.

A high-level flow diagram of the general approach to cloud layer segmentation is provided in Figure 1. Input parameters include pixel-level EDR information such as the cloud top height and optionally cloud effective particle size and cloud optical depth. The pixel-level cloud mask and confidence flags are used as well. The first stage in the algorithm is a correction for oblique views. Pixel ground locations are adjusted to remove the parallax positioning error due to the viewing geometry. For cloud-covered pixels, a density estimation procedure is used to determine the degree to which nearby pixels are similar in EDR values to the pixel under consideration. From each pixel, the nearby pixel for which the density increases the most per specific distance is found and labeled the *predecessor* of the central pixel. Following the chain of predecessors from each pixel leads to a small set of root pixels, which are at local peaks of the density function and have no predecessors. The group of pixels whose chains lead to a given root pixel are considered to be members of the same cloud layer. All the pixels in each grid cell are considered in this manner. All distinct cloud layers with member pixels within a grid cell are listed. The fractional coverage of each layer within the cell is the un-obscured cloud fraction. The average pixel value of an EDR parameter (e.g., cloud top height) in a specified cloud layer can be assigned as the layer EDR value using the association between pixels and layers. In fact, this pairing is used to produce layer values of all cloud EDRs in a separate aggregation algorithm not described here. Some layers within a single grid cell will be statistically similar enough to be merged into a single layer. Finally, layers are ordered by their altitudes, and another statistical procedure is used to estimate the amount of each layer that is obscured by higher altitude clouds.

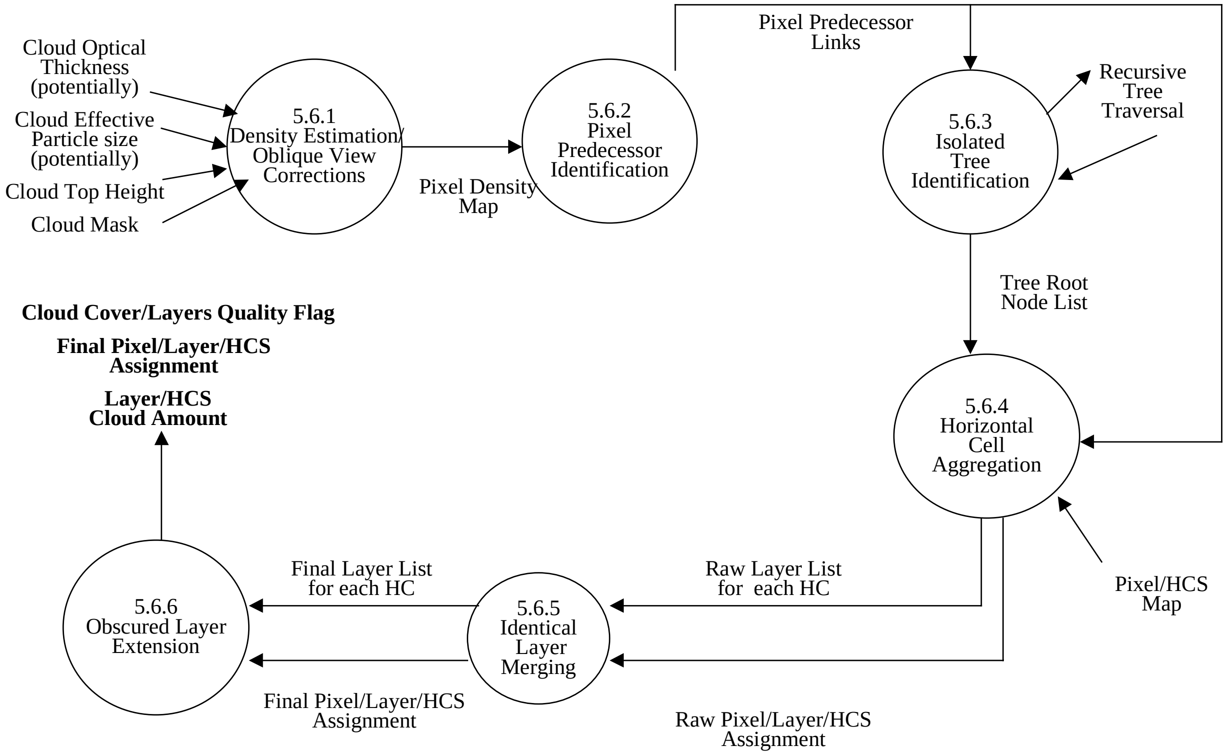
Cloud Optical Thickness (potentially)
Cloud Effective Particle size (potentially)
Cloud Top Height
Cloud Mask
5.6.1 Density Estimation/ Oblique View Corrections
Pixel Density Map
5.6.2 Pixel Predecessor Identification
Pixel Predecessor Links
5.6.3 Isolated Tree Identification
Recursive Tree Traversal
Tree Root Node List
5.6.4 Horizontal Cell Aggregation
Pixel/HCS Map
Raw Layer List for each HC
Raw Pixel/Layer/HCS Assignment
5.6.5 Identical Layer Merging
Final Layer List for each HC
Final Pixel/Layer/HCS Assignment
5.6.6 Obscured Layer Extension
Cloud Cover/Layers Quality Flag
Final Pixel/Layer/HCS Assignment
Layer/HCS Cloud Amount

Figure 1. High-level flow diagram.

Figure 2 shows a sample image generated using graph theoretic clustering. The clustered image (left) was generated from linear combinations of the GOES-8 infrared (IR) channels. Each cluster is displayed with the long-wave infrared (IR) intensity of its root pixel. Background clear pixels are displayed in solid black. To the right of the cluster image is the corresponding IR image from the same time.

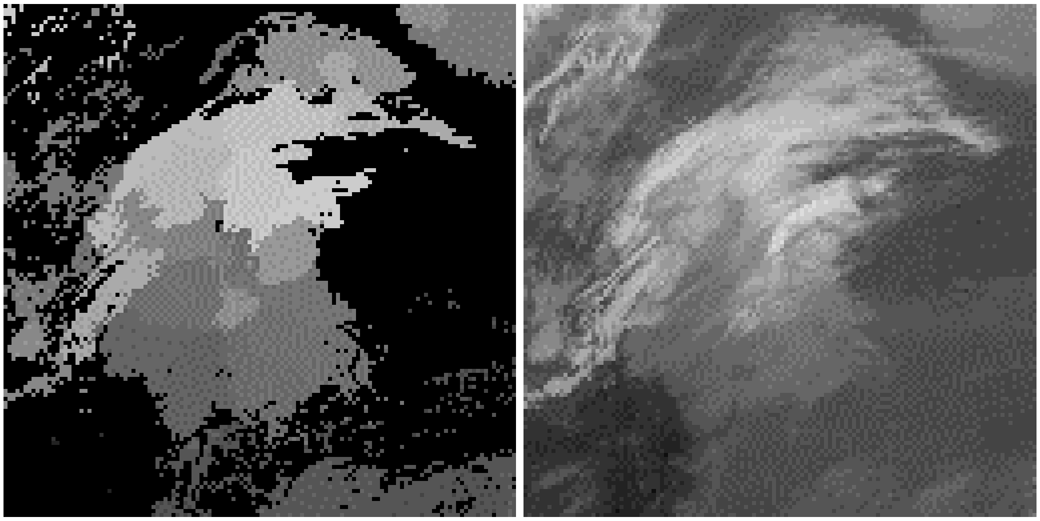

Figure 2. Sample graph theoretic clustered image and IR image.

3.1 PROCESSING OUTLINE

The following steps are executed in sequence:

3.1.1 Step 1: EDRs and Supporting Data Ingest and Preprocessing

Supporting EDRs may include, but are not limited to, the VIIRS Cloud Mask, Cloud Top Height, Cloud Optical Depth, and Cloud Effective Particle Size. Preprocessing involves using the Cloud Top Height in conjunction with the satellite viewing geometry to correct the locations of pixels for the oblique viewing effect, also known as the parallax error. Each pixel along a scan is considered in order of increasing Cloud Top Height. The elevation angle to the viewing sensor and the Cloud Top Height are used to determine the parallax error of positioning for the given pixel. The pixel is then relocated by the parallax error towards the sensor; a position appropriate for a nadir view. Subsequent higher altitude pixels may obscure earlier, lower altitude pixels, causing cloud cover estimates to decrease. All input EDRs are relocated along with the corresponding pixels.

3.1.2 Step 2: Density Estimation

The density at each pixel is a measure of the degree of similarity of the physical characteristics of the pixel to the surrounding pixels. *Density* is the proper, formal term used in the pattern recognition literature. *Similarity* is used herein as a more readily ponderable term having the same definition and meaning. Wherever neighboring pixels vary only slightly in EDR values, the similarity (and identically, the density) measure is high. Wherever the EDR values of neighboring pixels are dramatically different, the similarity (density) measure is low. The local density space forms a “contour map” of the *landscape* across an image.

For example, consider the region within a horizontal cell. Assume this region is evenly divided between two homogenous layers: low (warm) clouds on the left and high (cold) clouds on the right. Furthermore, assume that the next horizontal cell to the left contains only low clouds, and the cell to the right contains only high clouds. At the far left of the central horizontal cell, we compute the density. Since all the nearby pixels have exactly the same cloud top height, the local similarity is a maximum. Nearer to the boundary, but still on the left, the density begins to drop. This happens because the pixels at and across the boundary have different cloud top heights from the location where we are computing the density. Therefore, the density drops as we approach the boundary between the two clouds layers. The density gradient near the boundary is directed away from the boundary–the gradient being the direction the density increases the most in.

The density for a pixel is determined by integrating the region around the pixel with a Gaussian weighting function. The region is defined not only by spatial proximity, but also by proximity in the EDR space. Empirically determined weights are used to sum the squares of the spatial distances and the squares of the differences of EDR values into a single squared distance term. The sum of Gaussian distance weights for all the pixels around the central pixel is the density for the central pixel.

3.1.3 Step 3: Pixel Predecessor Identification

The predecessor of the pixel is the neighboring pixel to which the local similarity increases the most. This predecessor is directly “uphill” in the density space. One or more pixels connected to the same predecessor forms a branch of a tree relationship. These relationships can be identified in succession up slope on a given hill in density space.

Continuing the example above, the pixel immediately to the left of the boundary has the next pixel further to the left assigned to be its predecessor. The density gradient directed to the left, and this was the closest pixel on that side. In turn, that pixel took its left-hand neighbor as its predecessor and so on, until a pixel was found where the density was greater than at any other neighboring pixel. This pixel may not be in the same cell initially considered; clearly, another cloud layer boundary must exist somewhere further to the left, possibly not in the original horizontal cell.

The predecessor of a pixel is found by searching a small neighborhood around the pixel. Ideally, the chosen predecessor is directly “uphill” from the central pixel in terms of the density. In practice, the pixel closest to being directly uphill is chosen. This is determined by estimating the spatial slope of the density from the central pixel to each candidate predecessor. The predecessor with the highest slope (provided the slope is greater than zero) is selected to be the actual predecessor. In some cases, no pixel can be found with a slope greater than zero. In these cases, the central pixel is assigned to be its own predecessor, i.e., it is determined to be a root pixel. The relationship between a pixel and its predecessor is a directed path from the former to the latter.

At this point, every pixel is either a root pixel, has a root pixel assigned as its predecessor, or has another pixel assigned as its predecessor. Because this is a gradient-following algorithm, the directed paths from pixels to predecessors do not circulate but only converge. Except for the trivial case of root pixels, no pixel can trace a series of directed paths back to itself, but only to a root pixel.

3.1.4 Step 4: Isolated Tree Identification

The root of an isolated tree is the pixel located at the top of the local hill in density space. A tree is termed *isolated* when no other tree is connected to it. The valleys between hills in the density space are the boundaries between isolated trees.

A unique cluster identifier is assigned to each pixel known to be a root pixel. For all other pixels, their paths are traced to the appropriate root pixel. The identifier for the root pixel is then propagated back through the series of directed paths to the original pixel, and is recorded at each pixel along the route. In this manner, every pixel is assigned a cluster identifier, and any pixels sharing the same cluster identifier have directed paths leading to the same root pixel. All pixels with the same identifier are members of the same cluster.

In our example, the pixel with the maximum density on each side of the boundary becomes the root pixel for each analyzed cloud layer. From each root pixel, we follow the predecessor links backwards, recording each pixel as being in the layer defined by the root pixel, until we find a pixel which is not the predecessor of any other pixel. That pixel lies on the outer boundary of the layer.

3.1.5 Step 5: Horizontal Cell Aggregation

Aggregation summarizes the layers formed within a horizontal cell by the set of distinct hills in density space that cover the horizontal cell.

All clusters that have member pixels within a given grid cell are recorded for that cell. The value for each EDR input into the algorithm (cloud top height and optionally cloud optical depth and effective particle size) for each layer in the grid cell is determined by averaging the pixel level EDRs of the member pixels. The fractional coverage of each layer is the fraction of all pixels in the grid cell that are members of the layer. The total cloud amount for the grid cell is the fraction of all pixels in the grid cell that are members of any layer.

In the example, on the left side of the cell all the pixels within the horizontal cell assigned to the left root pixel would be counted, and the mean and variance of their cloud top heights would be found. These statistics would be recorded for the low cloud layer in this horizontal cell. The pixels assigned to the right root pixel would be counted and mean and variance of the pixel cloud top heights would be computed for the high cloud layer within the horizontal cell.

3.1.6 Step 6: Identical Layer Merging

Many neighboring hills in the density space are, in fact, statistically (and possibly perceptually) identical. The identical layer merge process identifies these cases within each horizontal cell and merges them together. This process can be thought of as statistically identifying small, isolated, *twigs* and appending them to nearby branches.

The layer merging process looks for small extent layers that can be absorbed into larger extent layers without substantially changing the EDR statistics of the larger layer. When a pairing of this type is attempted, the Bhattacharyya distance (a measure of the dissimilarity of two distributions) is computed for the larger layer before and after absorbing a smaller layer. For a given small layer, the larger layer with the shortest distance is considered the best merge candidate. If the distance for the best candidate is less than an empirically determined threshold, the pixels in the smaller layer are reassigned to the larger layer, and grid cell Cloud Cover/Layers EDRs are recomputed.

Continuing with the example, the cloud top height means and variances of the two would be used to compute the Bhattacharyya distance between the two layers' cloud top height distributions, and the unlikelihood of accidentally measuring two distributions with this distance is computed. If the unlikelihood is high enough, the layers are retained as separate entities. If the likelihood were very reasonable, then we would assert that the layering was arbitrary, and merge all the pixels into a single layer.

3.1.7 Step 7: Obscured Layer Extension

Obscured layer extension attempts to estimate the amount of lower altitude layers that are likely to be covered by higher altitude layers under the assumption that the distribution of cloud cover inside horizontal cells is uncorrelated between layers. This assumption is only made inside aggregation horizontal cells, and does not in any way imply that clouds in different layers do not correlate over larger scales. An assumption here of lack of correlation is a reasonable first guess. Correlations within horizontal cells that do exist are generally due to terrain effects that extend up to high altitudes (e.g., air rising over a mountain to form mountain cap clouds). On other occasions reverse correlations may be present. A constant assumption of lack of correlation will split these cases and provide a reasonable result for the bulk of cases when strong effects are not present.

It is likely that layers may be partially obscured by higher altitude layers. A statistical method can be used to estimate how much of a lower layer cannot be sensed if an assumption of the degree of correlation between the two layers can be made. To simplify the task, an assumption of zero correlation is made, i.e., that cloudiness in one layer does not predict cloudiness in another. In this case, a simple formula is used to estimate the amount of obscured coverage in each lower layer in turn. For each potentially obscured layer, the obscured coverage is estimated from the unobscured coverage of the layer and the coverages of other layers both above and below it.

The resulting value from the formula is stored as the estimated obscured coverage of the layer. The total cloud cover is unaffected by these computations.

In the example, no pixels were found without clouds. In this special case, all the pixels in which there were high clouds are also assumed to have hidden low clouds beneath them. With the assumption that the coverage of the two layers within the grid cell is uncorrelated, it is unreasonable to assume there are not low clouds underneath. After all, for every pixel where high clouds were not in the way, low clouds were detected.

3.2 ALGORITHM INPUTS

3.2.1 VIIRS Data

The Cloud Cover/Layers EDR uses other VIIRS EDRs as its primary inputs. The accuracy of the retrieval of these other EDRs will directly impact the performance of the layered cloud retrieval. Gross errors (e.g., spatially consistent, large magnitude errors) in these other parameters will have little effect on the layered cloud retrieval algorithm due to the robust statistical data clustering algorithm used to aggregate pixels into layers. Entirely random errors (e.g., instrument noise) will have a slight impact on the algorithm, but will be mitigated by the action of the clustering algorithm as a noise filter. The most significant impact on effective layering is the presence of spatially correlated errors. These errors may be interpreted by the algorithm as breaks in otherwise contiguous clusters, or even as entirely distinct, erroneous clusters consisting only of errors from the input data. Therefore, the essential characteristic for all of the input EDRs is spatial consistency.

The EDRs that will be relied upon are Cloud Top Height and optionally Cloud Optical Depth and and Cloud Effective Particle Size. Although these EDRs will be available on the aggregation grid, the Cloud Cover/Layers Algorithm requires them on a per-VIIRS pixel basis. In addition, the cloud cover/layers process requires the VIIRS Cloud which is used to identify and distinguish cloudy pixels from clear pixels.

3.2.1.1 Cloud Mask and Mask Diagnostics

The VIIRS cloud mask will provide the fundamental spatial depiction of clouds. Only those pixels identified as being cloudy will be processed by the Cloud Cover/Layer EDR algorithm.

3.2.1.2 Cloud Top Height

Cloud Top Height serves as the primary physical characteristic used to distinguish layers. The predecessor trees constructed from the available EDRs are intended to rely heavily on the Cloud Top Height. Clouds, which stratify at different altitudes, are almost universally accepted to be in different layers, regardless of other, more superficial similarities. The obscured layer extension may not be performed without knowledge of the vertical order of layers, which is only available through the cloud top parameters. The cloud layer analysis will be severely degraded without the cloud top data.

3.2.1.3 Cloud Optical Thickness

In a further distinction after cloud top height, thin clouds may be considered distinct from opaque clouds. Thus, despite perceptual inclinations to the contrary, thin edges of cirrus layers may be classified as layers separate from the main mass of cirrus clouds.

3.2.1.4 Cloud Effective Particle Size

Finally, effective particle size may also be of use in distinguishing layers that occur at the same altitudes but which consist of different cloud types.

3.2.1.5 Other Pixel EDRs (TBD)

Other EDRs may be ingested into the layer algorithm depending on ultimate degrees of quality achieved. For example, the Cloud Base Height EDR, where available, would clearly be of equal importance to the cloud top heights.

3.2.1.6 Other Pixel Radiances (TBD)

In some exceptional cases, raw radiance values may be ingested. This might be the case whenever actually recovered EDRs were not available due to some systematic processing failure.

3.2.2 Non-VIIRS Data

Some constants and data not derived from the VIIRS instrument will be required. User-specified control parameters will include the following.

3.2.2.1 Neighborhood Density Kernel Radius

The local density is a function of the degree of similarity of a pixel with its neighbors. The density kernel radius defines how close together pixels must be to be considered neighbors.

3.2.2.2 Neighborhood Gradient Search Radius

When predecessors are identified, only pixels within this distance will be considered.

3.2.2.3 EDR Density Scaling Factors

The EDRs, which contribute to the local density function, are first scaled relative to these amounts.

3.2.2.4 Bhattacharyya Layer Merge Test Significance Threshold

The Bhattacharyya test compares the degree of overlap of two multivariate Gaussian distributions. The significance threshold indicates how obviously different the distributions of physical characteristics of two layers must be before they are not merged into a single layer.

3.2.2.5 Interlayer Correlations

Interlayer correlations are to be set to zero, unless some physical rationale for a non-zero value is present. A non-zero value will either preferentially extend an obscured layer underneath other layers (for positive correlations) or avoid extending layers underneath others (for negative correlations).

3.2.2.6 Land/Sea, Terrain Data, Snow/Ice Flags, etc.

These data are presumed to have been used in the recovery of the VIIRS parameters and are thus used implicitly in the EDRs and the VIIRS cloud mask.

3.3 Theoretical Description

3.3.1 Physics of the Problem

Rather than being a retrieval in the conventional sense of using a physical model to recover unknown parameters from measurements, the layer specification algorithm is statistical in nature and is more involved with perceptions and degrees of belief in consistency.

3.3.2 Mathematical Description of the Algorithm

The cloud cover/layers algorithm consists of four main components: the Oblique View Correction, the Graph Theoretic Clustering (GTC), the Layer Merge Test (LMT), and the Obscured Layer Extension (OLE). The mathematical basis of each of these algorithms will be examined in turn.

3.3.2.1 Oblique View Correction

An oblique view correction is performed to correct observed pixel locations for parallax viewing errors.

3.3.2.2 Graph Theoretical Clustering

Graph theoretical clustering is a robust, efficient, and noniterative clustering technique designed to operate on sparse, randomly distributed data. It consists of density estimation, predecessor identification, and graph tree isolation steps.

Density estimation

The density estimation algorithm determines the degree of similarity between pixels in a small region. The region is defined in terms of both geographic and parameter spaces. Although the geographic space is regularly spaced rather than randomly distributed, the nonregularly spaced measurements in parameter space meet the requirements of the GTC algorithm for randomly distributed data. This is, perhaps, weaker compliance than originally envisioned, but the algorithm is sufficiently robust to perform well.

Density estimation is performed by computing the convolution of a distance weighting kernel and the local distribution of pixels in combined geographic and parameter space:

$$\rho = \iint_R c \exp\left[-\frac{1}{2}\left(\frac{d^2}{L^2} + \sum \frac{\Delta e^2}{w^2} \right) \right] dA \tag{1}$$

where ρ is the local density, R is the local region, c is the cloud mask cloud flag (1=cloud, 0 = clear), d is the distance from a pixel to the center of the region, L is the density kernel radius, Δe is the difference in each EDR from the pixel to the central pixel, w is the EDR scaling factor, and dA is the differential area. Naturally, the integral is computed with a discrete approximation.

Predecessor identification

Predecessor identification is performed by determining the geographically neighboring pixel in the direction of the highest density gradient. Over a limited search zone, the pixel with the highest gradient-to-distance ratio is identified. The gradient is estimated by the quotient of the change of the density and the intervening distance:

$$\left(\frac{\Delta \rho}{d} \right)_i = \frac{\rho_i - \rho_0}{d_{i,0}} \tag{2}$$

where ρ_0 is the density at the point in question, ρ_i is the density at another location i, and $d_{i,0}$ is the distance between the two points. In the local neighborhood defined by all pixels within the neighborhood search radius, the point i is found that maximizes the gradient. That central point has the point i recorded as its predecessor.

From calculus, it is well known that the gradient of a scalar field is a vector field of pure divergence. This property ensures that the chain of predecessors does not circulate, but instead leads uniquely to a single ultimate predecessor that has no other predecessor than itself.

Isolated tree identification

Isolated tree identification is performed by identifying the set of pixels whose predecessors converge to the same ultimate predecessor. A recursively defined search algorithm propagates the identity of each ultimate root predecessor out through the tree branches to all leaf pixels whose predecessor chains lead to the same root.

3.3.2.3 Layer Merging Test

The Layer Merging Test is used to identify groups of cloud-covered pixels that are sufficiently similar to be considered in the same cloud layer.

Bhattacharyya test statistic

The Bhattacharyya distance measures the underlap between two multivariate Gaussian distributions. Under the assumption of equal priors, this measure estimates the error rate of a two-class discrimination problem. If the distance is short, the classification error rate will be very high; if the distance is great, the error rate will be very low. The Bhattacharyya metric is

$$\mu = \frac{1}{8}\left(\mathbf{M}_2 - \mathbf{M}_1\right)^T \left[\frac{\Sigma_1 + \Sigma_2}{2}\right]^{-1} \left(\mathbf{M}_2 - \mathbf{M}_1\right) + \frac{1}{2} ln \frac{\left|\frac{\Sigma_1 + \Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}} \qquad (3)$$

where M and Σ are the mean vector and covariance matrix of the two multivariate Gaussian distributions being compared. For example, the data elements being considered could be the cloud top height/temperature, the cloud optical depth, and the cloud effective particle size.

The Bhattacharyya test examines two sampled multivariate distributions and answers this question:

If the smaller population sample (sample 1) were added to the larger population sample (sample 2) to form a composite sample (sample C), what would be the likelihood that samples of size equal to sample 2 and sample C could have been drawn at random from the same population and have as large a Bhattacharyya distance?

If this probability is very low, then we say that sample 1 should not be combined with sample 2 (because it changes sample 2 too much). If the probability is moderate or large, then we say that sample 1 may be combined with sample 2 (because it is compatible with sample 2).

Significance estimation

The distribution of the Bhattacharyya distance under the null hypothesis is not known to have been derived analytically, but can easily be estimated through Monte Carlo simulation. The null hypothesis for the statistic states that both samples were drawn from the same parent population. Given the number of counts in each sample, the cumulative probability that a Bhattacharyya distance less than or equal to the measured value is the significance of the statistic. Clearly, for a sufficiently large distance value, the significance will be near 1, indicating a very unlikely distance under the null hypothesis. For very small distances, the significance will be near zero, indicating high compatibility with the null hypothesis.

Merge criteria

The algorithm requires a significance decision threshold, which will be used to determine which layers will be merged. The user specifies a degree of significance for merging layers. This user-specified value must be chosen to balance merging layers that are in fact different, and not merging layers that are in fact the same.

The merge process is executed for each aggregation horizontal cell in order of layer population. The smallest population layer is combined with each higher population layer on a trial basis. If the Bhattacharyya test statistic between each larger population layer and the trial composite layer is small, this indicates that the smaller layer might plausibly be combined with the larger layer without changing the fundamental character of the larger layer. The smaller layer is merged into the larger layer with the smallest test significance if the significance is below the user-supplied threshold.

3.3.2.4 Obscured Layer Extension

Layers within a single horizontal cell may overlap. Statistical properties can be used to estimate the overlap. The VIIRS sensor images the surfaces of clouds that are in view of the sensor. When a layer is wholly covered by another layer, it will obviously not be detected. However, for those layers that protrude from underneath a higher obscuring layer, statistical techniques can be used to estimate the obscured fraction, using a Layer Coverage Correlation Model.

Given two layers, one obscuring the other within a horizontal cell, there are only four possible combinations for each pixel in the horizontal cell: neither layer is present; both layers are present; the high layer alone is present; and the lower layer alone is present. Knowing the true state of natural cloud coverage within the horizontal cell for both layers, we can easily compute the correlation between layers using the first approximation to the tetrachoric correlation.

The approximation to the tetrachoric correlation is given by:

$$\rho \approx \sin\left(\frac{\pi}{2} \frac{\sqrt{AC} - \sqrt{BD}}{\sqrt{AC} + \sqrt{BD}} \right) \tag{4}$$

where A is the joint frequency of X and Y, C is the joint frequency of (NOT X) and (NOT Y), B is the joint frequency of X and (NOT Y), and D is the joint frequency of (NOT X) and Y. Both X and Y are possibly non-independent Bernoulli random variables.

The approximation of the tetrachoric correlation is computed from knowing the frequencies of occurrence of the four possible combinations of the true state of nature. If we are given any three of the combinations and their correlation, we can easily solve for the missing combination's frequency.

3.3.2.5 Extension Function

The extension function assumes a correlation and inverts the model to recover the true layer extents. Given knowledge of the correlation beforehand, the correlation function can be solved for any other parameter:

$$\rho \approx \sin\left(\frac{\pi}{2}\frac{\sqrt{C(U-L)}-\sqrt{UL}}{\sqrt{C(U-L)}+\sqrt{UL}}\right) \tag{5}$$

where C is the fractional amount of coverage in a layer, U is the total fractional coverage of all higher layers, and L is the amount of obscured coverage of the layer. Assuming equality, we can solve for L:

$$L = \frac{CU(1+\alpha)^2}{(1-C-U)(1-\alpha)^2 + C} \tag{6}$$

where

$$\alpha = \frac{2}{\pi}\arcsin(\rho) \tag{7}$$

For layer extension, the correlation is assumed to be zero, and the function is solved for the coverage of the lower of the two layers. This process is executed successively for each layer from the top down:

$$L = \frac{CU}{1-U} \tag{8}$$

One can easily see that L must always be between 0 and U.

3.3.3 Archived Algorithm Output

None.

3.3.4 Variance and Uncertainty Estimates

Errors in the Cloud Cover/Layers EDR arise from several sources and in several places. Individual algorithms are sensitive to additive noise, band-to-band registration errors, and other effects. Table 5 summarizes possible error sources.

Table 2. Algorithms and error sources.

Algorithm	Additive Noise	Registration Errors	Others
Graph Theoretic Clustering	Yes, Section 3.3.4.2	Yes, Section 3.3.4.2	No

Layer Merging Test	Minimal, Section 3.3.4.3	Minimal or No Effect	Non-Gaussian Distributions, Section 3.3.4.3 Significance Estimation, Section 3.3.4.3 Merge Criterion, Section 3.3.4.3
Obscured Layer Extension	Minimal or No Effect	Minimal or No Effect	Layer Coverage Correlation Model, Section3.3.4.4 Extension Function Sensitivities, Section 3.3.4.4

3.3.4.1 Error Budget

For a complete description of the errors that impact the Cloud Cover/Layers Algorithm see the Raytheon VIIRS Error Budget, Version 3 (Y3249). Those error budgets are predicated on the linearity and independence of errors. In the Cloud Cover/Layers Algorithm, the contributing components are strongly coupled (i.e., non-independent) and act nonlinearly. Therefore, an explicit error budget does not fully describe this EDR. Following is a qualitative description of the sources and interactions of errors within the CC/L Algorithm.

Graph theoretical clustering

Graph Theoretic Clustering acts on the input data to form small clusters of pixels which share increasing similarity from the outer boundary toward the topological center of the cluster. Several sources of error affect this statistical clustering algorithm. Density (or similarity) errors may result from both registration and additive noise.

- *Registration Errors*. Band-to-band and absolute registration errors displace the parameter dimensions relative to each other and to the geographic dimensions in the density space. As long as the errors are orders of magnitude smaller than either the density-weighting kernel radius or the neighborhood-gradient search radius, the effects will be very small. These user parameters determine the scale of the Cloud Cover/Layers Algorithm as a data filter, and hence are innately robust in the presence of registration errors.

  Absolute registration errors of the ensemble of parameter dimensions with respect to the geographic coordinates displace pixels out of the true geographic horizontal cells to which they belong into neighboring cells. These displacements will be very small so long as absolute registration errors are orders of magnitude smaller than the horizontal cells.

- *Additive Noise Effects*. As the fundamental action of the algorithm is to act as a filter, the computation of the density function for a given pixel is not biased unless the additive noise has a spectral signature with characteristic scale larger than the size of the filter effect. Given uncorrelated (white) additive noise, no biasing effect will occur.

The cloud mask can be very sensitive to additive noise in the thermal channels. As the cloud mask is used to explicitly compute the amount of cloud in each layer, a direct relation will hold: errors in the cloud mask are propagated 1 for 1 to the layered analysis.

Predecessor identification and isolated tree identification

The magnitudes of layer population errors given errors in density are very small. Each layer represents a single hill in the density space. On the slope of each hill, errors in density estimates do not change the population of pixels assigned to a layer. Only errors near the boundaries between hills can result in miss-assigned pixels. The boundaries are defined as the locus of points between hills where the density gradient is identically zero. Near these boundaries, the density gradient is likely to be small in magnitude, if not exactly zero, and relatively more susceptible to error propagation. Hence, population errors are likely to be proportional to the boundary length of each layer. Therefore, error is proportional to the square root of the population in each layer. The layers with the greatest cloud amount will have the least estimation error in the presence of input errors. For layers large enough to include large portions of the cell’s edges as boundaries, few if any errors in magnitude result regardless of noise content.

Layer merging test

Substantial additive noise will broaden and circularize the nominal distributions being compared. When only a small fraction of the total variance is due to white noise, the test statistic maintains its meaning and degrades smoothly as more noise is added. For some peculiar distributions (e.g., those that are nearly degenerate), the addition of a small amount of Gaussian noise actually improves the performance of the test.

The Bhattacharyya test relies heavily on the multivariate normal assumption. If any parent distribution (the underlying true distributions from which samples are taken) is non-normal along any principal axis, or if it exhibits a complex correlation structure, then the Gaussian assumption breaks down, and the test will not work well. In the context of combining layers on the basis of similarity of distributions of EDRs, the normal assumptions are very likely to hold.

Significance estimation

The significance function was empirically fit to the results of Monte Carlo simulations. For all simulation cases, enough independent runs were made to drive absolute error below 1 percent (i.e., the estimated significance level would always be within 1 percent of the true value). However, relative error margins can be very high in some cases, particularly when very few samples (e.g., < 10) or very many samples (e.g., >100,000) are contained in either distribution. When any reported significance level is close to zero, the relative error can be very large. However, given the 1 percent bound on absolute error above, this should not pose a practical limitation.

Merge criterion

The merge process requires that a threshold significance level be provided. The majority of merge cases have either very high or very low significance levels. If the threshold is in the middle ground (i.e., near one half), it is less likely to result in a misclassification.

Obscured layer extension

The correlation between cloud layers within a horizontal cell is assumed to be zero. This correlation is the tetrachoric correlation of the presence of cloud in a given layer with the presence of cloud in any higher altitude layer within the geographic area of a horizontal cell. There are circumstances where clouds are obviously and strongly correlated between layers. For example, mountain wave clouds tend to form at multiple levels and in almost exactly the same locations at each level. Large geographically distributed weather systems also correlate cloud cover between altitudes, but infrequently within a single horizontal cell. The small scale state and field fluctuations that cause partial cloud cover to distribute in a particular fashion within a small horizontal cell do not tend to extend across many levels of the atmosphere. This assumption is used to assert that layered cloud presence tends to be uncorrelated within horizontal cells.

Derivation of errors resulting from perturbations in the visible layer extends to the assumed correlation. Differentiating the layer extension function with respect to correlation shows us the sensitivity of the amount of cloud to perturbations in the correlation. Under the assumption that the correlation is not large (i.e., <1/2), the sensitivity is linear over the correlation and roughly proportional to one-half the correlation. A 10 percent error in the correlation corresponds to a 5 percent error in amount of extended cloud.

3.4 ALGORITHM SENSITIVITY STUDIES

3.4.1 Radiometric Calibration Errors

The algorithm, being almost entirely statistical in nature, is insensitive to band calibration errors. Systematic errors in EDRs feeding the layered cloud algorithm resulting from miscalibration are unlikely to result in significant layered cloud errors due to the robust statistical process (Graph Theoretic Clustering) being employed. Only variations in results directly attributable to errors in the input EDRs should be expected. For example, the total sum of cloud coverage in all layers might be in error, but only because the cloud mask was in error.

3.4.2 Instrument Noise

The computation of density in the GTC algorithm acts like a low pass filter of the same radius, and is therefore insensitive to instrument noise propagated through the other EDRs.

3.4.3 Other

None.

3.5 Practical Considerations

3.5.1 Numerical Computation Considerations

In the CC/L algorithm, numerical approximations to various statistical tests are performed. None of these, however, have unstable numerical properties nor are they applied in iterated fashion, so as to amplify numerical errors. Therefore, the CC/L algorithm is resistant to numerical problems associated with finite precision arithmetic on computers.

3.5.1.1 Density Field and Gradient Estimations

In the GTC algorithm, as adapted for imagery data, the density is the probability mass of pixels that are close (as defined by the distance kernel) both in geographic space and in parameter space.

The discrete nature of the pixel data, as opposed to a continuous field of values, imposes sampling precision limits on the density measure. On this account, the neighboring region over which the density is computed must be sufficiently large so that the true density and not merely random error components can be measured.

3.5.1.2 Restricted Neighborhood Search

In the general GTC algorithm, a small neighborhood is searched to identify the predecessor pixel to which the gradient is greatest. Because the image data in the adapted GTC algorithm is regularly spaced, smaller neighborhoods may be used for the predecessor pixel. Were pixels placed randomly in geographic space, a larger number would need to be examined to ensure that the cardinal directions were all examined. With rectangularly arrayed pixels, the cardinal directions are efficiently covered by a very small search set. For example, a rectangular search area of distance equal to two pixels covers 16 unique direction vectors outward from the center pixel.

3.5.1.3 Bhattacharyya Test Estimation

The Bhattacharyya test significance level is empirically fit from Monte Carlo simulations, and is accurate to 1 percent absolute.

3.5.2 Algorithm Testing

Testing of the algorithm is being performed in conjunction with algorithm calibration to synthetic *ground truth* cases run through the Cloud Scene Simulation Model (CSSM). The input algorithm parameters are set to calibrate the algorithm to agree with a known set of results, and then the algorithm is tested against another independent set of test cases to demonstrate continued correct performance.

3.5.3 Diagnostics

A series of stratification flags indicate conditions that might have an impact on quality.

A series of measures relating to algorithm processing (e.g., total number of layers generated per horizontal cell, average altitude of highest layer) are computed and saved for review.

3.5.4 Quality Control

For operations, quality control procedures will be required for this algorithm on a continuing basis.

3.5.4.1 Periodic Analyst Review

A trained and experienced analyst directly compares algorithm outputs with inputs and records problems.

3.5.4.2 Comparison with Trusted Observations

Some trusted conventional observations (e.g., daytime synoptic observations from developed-world stations), coincident in time and space with the layered cloud depiction, are used as proxies for ground truth.

3.5.4.3 Statistical Change Detection

Diagnostic quantities are fed into a statistical change detection algorithm over time. When a significant set of recent diagnostic values fall out of compatibility with previous values, a flag is raised. This event detects intervention in the process (i.e., a previously stable process has been altered by an exterior event), and is followed up with more detailed investigation by an analyst.

3.5.5 Exception Handling

The CC/L algorithm implementation was designed to handle exceptional conditions without requiring external intervention. Missing and/or inconsistent data, division-by-zero, extreme parameter values (e.g., 1.0 or -1.0 correlations), singular (i.e., non positive-definite) covariance matrices, and other exceptional conditions are all trapped and properly handled.

3.5.5.1 Missing or Degraded EDR Data

Essential EDRs

A compact set of EDRs without which the algorithm may not yield threshold quality results. This set includes the Cloud Mask and the Cloud Top Height EDRs. The cloud mask is necessary to determine which pixels will be input to the layering algorithm. Without the cloud mask, the algorithm must assume all pixels are cloudy, and will overestimate cloud cover in proportion to the worldwide cloud-freeness. Cloud top height provides vertical extent information without which the layering algorithm could not discriminate between the most simple cloud layers (e.g., low, middle and high cloud types). The full list of essential EDRs may be expanded as further investigation continues.

Non-Essential EDRs

EDRs whose presence improves the layer analysis above threshold, but without which the threshold is still met. This set includes all other inputs not deemed essential through flowdown result analysis. This includes Effective Particle Size and Cloud Optical Thickness.

3.5.5.2 Degraded Processing Environment

N/A. The degraded operating environment of the CC/L is the degraded operating environment of the essential input EDRs, the VIIRS Cloud Mask and the Cloud Top Height EDRs.

3.5.5.3 Algorithm Runtime Scaling (Quality Performance Tradeoff Space)

The layer algorithm may be configured to run more quickly by changing the numerical approximations used for computing the density and density gradients.

3.5.5.4 Degradation of Temporal/Spatial Resolution (Scale Size Tradeoff Space)

Run-time is expected to be proportional to the number of pixels that must be processed. If pixels are aggregated into larger *super-pixels* prior to execution of the algorithm, performance will improve at the expense of the smallest extent layer that may be discerned. For example, if nominal imagery is a 0.4 km resolution, a 3x3 averaging process may be applied to reduce the number of pixels by a factor of 9, yielding a 9 to 1 improvement in runtime. However, the cost is the inability to resolve layers smaller than 1.2 km in extent.

3.6 ALGORITHM VALIDATION

The CC/L algorithm is best validated against MAS data and MOIDS data (when available) along the imagery track centers where lidar cloud profiling data is available. By inspection, a set of ground truth layered cloud amounts can be determined. These can then be compared to CC/L layered assessments and a qualitative indication of CC/L performance can be attained. Due to the lack of widespread ground-truth layered structure data, concrete measures of accuracy and precision cannot be determined. Determining these statistical measures of performance require a great deal of independent and representative cloud samples, which even the MAS and MODIS data may not provide.

4.0 ASSUMPTIONS AND LIMITATIONS

4.1 ASSUMPTIONS

4.1.1 Spatial Domain of Processing

An assumption of the appropriate domain of processing must be made. This assumption will dictate many characteristics of an operational production system and needs to be investigated further.

4.1.1.1 Large Scale Processing

Under this processing assumption, the computationally largest blocks of data are passed to the algorithm for processing. These data blocks may correspond to quarter-orbits of imagery-derived EDRs for a polar orbiting sensor, each covering a very large geographic region. Each stage of the algorithm is applied to the entire region at once.

4.1.1.2 Regional Processing

Smaller sections of the available imagery are processed in turn. These may correspond to blocks of scanlines received in short, real-time increments. Processing is confined to these regions, and slight discontinuities between regions may result.

4.1.1.3 Local Processing

Individual horizontal cells or small sets of horizontal cells are processed as entities. Maximal parallelization of processing may be achieved. Adjacent horizontal cells may not be entirely consistent.

4.1.2 Other

There are no other assumptions.

4.2 LIMITATIONS

4.2.1 Inherent Cluster Ambiguity

There is no best mathematical classification of data into clusters. Clustering is a fundamentally soft subject, where many different and equally justifiable interpretations of data are present. Any single partition of a data set is inherently ambiguous. Rescaling, rotations, arbitrary coordinate transforms, choices of distance metrics, selection of the cluster integrity measures, and other design decisions determine the performance of an algorithm.

4.2.2 Definition of Cloud Layers

In the introduction, various definitions of cloud layers were discussed. The choice of cloud layer definition dramatically affects the design and implementation of a cloud layer algorithm. This algorithm follows the vertically oriented slab model and is limited in modeling capability accordingly.

4.2.3 Other

There are no other limitations.

5.0 REFERENCES

5.1 General Pattern Recognition

Duda, O., and P.E. Hart (1973). *Pattern Classification and Scene Analysis*. New York: John Wiley & Sons.

Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. San Diego: Academic Press, Inc.

5.2 Specific to clouds

Bunting, T., and K. R. Hardy (1985). *Cloud Identification and Characterization from Satellites*. US Air Force Geophysics Laboratory, Hanscom Air Force Base, MA, AFGL-TR-85-0041, Chapter 6.

Garand, L. (1988). Automated Recognition of Oceanic Cloud Patterns. Part I: Methodology and Application to Cloud Climatology. *Journal of Climate,* January.

Harris, R. (1977). Automatic Analysis of Meteorological Satellite Imagery. *British Pattern Recognition Association and Remote Sensing Society Readings*, 45-72.

Toldalagi, P. M., and W. H. Lebow (1982). *Survey and Analysis of Satellite Cloud Classification Research Techniques*. NEPRF Contract No. 81-C-H155, Final Report, Naval Environmental Prediction Research Facility, Monterey, CA.

5.3 Numerical methods

Golub, G.H., and C. F. VanLoan (1989). *Matrix Computations*, 2nd Edition. Baltimore: Johns Hopkins University Press.

Kenny, J. F., and E. S. Keeping (1951). Tetrachoric Correlation. Number 8.5 in *Mathematics of Statistics, Pt. 2,* 2nd Edition. Princeton NJ: Van Nostrand, pp. 205-207.

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and P. P. Flannery (1992). *Numerical Recipes in C*, 2nd Edition. NY: Cambridge University Press.